

Self-Synchronized Duty-Cycling for Mobile Sensor Networks with Energy Harvesting Capabilities: A Swarm Intelligence Study

Hugo Hernández¹, Christian Blum¹, Martin Middendorf², Kai Ramsch² and Alexander Scheidler²

¹ALBCOM Research Group, Universitat Politècnica de Catalunya, Barcelona, Spain
{hhernandez, cblum}@lsi.upc.edu

²Department of Computer Science, University of Leipzig, Leipzig, Germany
{middendorf, kairamsch, scheidler}@informatik.uni-leipzig.de

Abstract—When asked if ants rest or if they work untiringly all day long, most people would probably respond that they had no idea. In fact, when watching the bustling life of an ant hill it is hard to imagine that ants take a rest from now and then. However, biologists discovered that ants rest quite a large fraction of their time. Surprisingly, not only single ants show alternate phases of resting and being active, but whole ant colonies exhibit synchronized activity phases that result from self-organization. Inspired by this self-synchronization behaviour of ant colonies, we develop a mechanism for self-synchronized duty-cycling in mobile sensor networks. In addition, we equip sensor nodes with energy harvesting capabilities such as, for example, solar cells. We show that the self-synchronization mechanism can be made adaptive depending on the available energy.

I. INTRODUCTION

In contrary to the general belief that ants are always busy, different species of ants have been observed to spend a large portion of their time resting; see, for example, [17], [9], [7], [4]. For example, ants of the species *L. acervorum* rest about 72% of their time. Moreover, not only individual ants present patterns of alternate activation, but also whole colonies show synchronized patterns of activity ([8], [14], [3]). In addition, activity phases are not just synchronized, but *self-synchronized* because no external signal has been found as a possible cause of colony synchronization. Delgado and Solé [5] modelled this behaviour of ant colonies by means of *fluid neural networks*. With their model they were able to show that synchronized activity pattern enable the colony to accomplish tasks more efficiently.

In this work we use the self-synchronization behaviour of ant colonies for the development of a self-synchronized duty-cycling mechanism for mobile sensor networks. Sensor networks ([19], [20]) aim to monitor large areas and to analyze complex phenomena for extended periods of time. Recent hardware advances produced sensors for a wide range of physical data such as light intensity, humidity, temperature and the oxygen level, as well as for the characteristics of objects such as direction and speed. This means that sensor networks can be used for many different tasks such as environmental monitoring, patient monitoring in health care, industrial machinery surveillance, etc. Some of these applications require the nodes of a sensor network to be distributed within wide areas without power sources, as, for

example, forests and seas. Moreover, sensor nodes might be mobile. Therefore, they are often equipped with batteries, which makes energy a scarce resource.

Several approaches can be found in the literature for extending the lifetime of a sensor network that is subject to energy limitations. A rather recent approach is referred to as *energy harvesting* (see, for example, [15], [12], [11]). The idea is to transform light or vibrations into energy that can be used to recharge the batteries. However, energy harvesting alone might not be enough for obtaining a sufficiently long network life time. An approach that aims at saving energy is *duty-cycling* (see, for example, [2], [6]). Hereby, sensor nodes periodically switch between energy intensive states and low energy states. Nodes in energy intensive states can perform all normal duties of a sensor node, whereas nodes in low energy states are restricted to certain functions in order to save energy.

Recently, some researchers made an attempt at combining dynamic duty-cycling with energy harvesting capabilities. Most of these works (see, for example, [10], [13]) require an apriori known energy profile. Such techniques require the energy source to exhibit little variations, which is, in many occasions, not very realistic. A recent work from Vigorito et al. [18] considers adaptive control techniques for adjusting duty-cycling without any apriori given energy profile. The main disadvantage of this approach is that it still considers each node separately and, although better performance and network lifetime is obtained, the lack of synchronizity between the nodes may imply that the network is restricted to certain applications. This is where the approach that we present in this paper comes into play. As mentioned before, our approach offers adaptive, self-synchronized duty-cycling.

Finally, let us remark that the study presented in this work is done entirely from the swarm intelligence point of view. It can be seen as a first feasibility study. We are perfectly aware of the fact that the proposed mechanism might be required to change when being adapted to real sensor networks.

The outline of this work is as follows. In Section 2 we shortly outline the model of ants' self-synchronization behaviour. In Section 3 we propose a first system together with

experimental results, whereas Section 4 presents a system with a slightly different behaviour, again accompanied by results. Finally, in Section 5 we offer conclusions and an outlook to future work.

II. SIMULATING ANTS' SELF-SYNCHRONIZATION BEHAVIOUR

Delgado and Solé [5] simulated the self-synchronization behaviour of ant colonies by means of *fluid neural networks* (FNN) [1]. The FNN proposed in [5] consists of a set of k automata or so-called *neuron-ants*. At each time step $t \in \mathbb{N}$, the state of each automaton $i \in \{1, \dots, k\}$ is described through a continuous state variable $S_i(t) \in \mathbb{R}$. Depending on $S_i(t)$, an automaton i is either *active* or *inactive*. This is expressed by the binary variable $a_i(t)$. When $a_i(t) = 1$ the automaton i is active, and inactive otherwise. The value for these binary variables is determined as follows:

$$a_i(t) := \Phi(S_i(t) - \theta_{\text{act}}) , \quad (1)$$

where θ_{act} is the so-called activation threshold, and $\Phi(x) = 1$ if $x \geq 0$, and $\Phi(x) = 0$ otherwise.

Automata are located on the nodes of a two-dimensional $L \times L$ grid with periodic boundary conditions. At each time step, active automata may decide to stay where they are, or they may move to one of the four neighbor sites with equal probability. Grid sites may be occupied by more than one automaton. Moreover, at each time step the state variable of each automaton is updated as follows:

$$S_i(t+1) := \tanh[gh_i(t)] , \quad (2)$$

where g is a gain parameter and $h_i(t)$ can be defined in diverse ways to obtain different behaviours. Note that the value of g determines how fast the value of variable $S_i(t)$ diminishes. Once an automaton becomes inactive, it can return to the active state either due to local interactions (see Eq. 2) or spontaneously with probability p_a and activity level S_a .

Finally, at each time step t the mean activity of the system is measured as follows:

$$A(t) := \frac{1}{k} \sum_{j=1}^k a_j(t) \in [0, 1] \quad (3)$$

Note that, the greater $A(t)$ the more automata are active at time t . Self-synchronization behaviour is characterized by an oscillating value of $A(t)$ over time; see, for example, Figure 1. The authors of [5] experimentally showed that they were able to obtain self-synchronization behaviour if the term $h_i(t)$ of Eq. 2 included interactions with automata that are located near automaton i :

$$h_i(t) := S_i(t) + \sum_{j \in N_i(t)} S_j(t) , \quad (4)$$

where $N_i(t)$ is the set of automata that are located in one of the eight nearest neighbour sites on the grid. Moreover, in order to obtain the desired behaviour they used the following parameter settings: $g = 0.1$, $\theta_{\text{act}} = 10^{-16}$, $S_a = 0.01$,

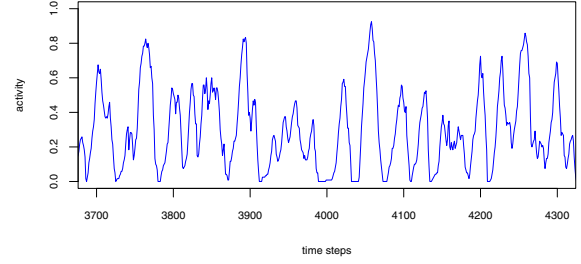


Fig. 1. Evolution of the system activity ($A(t)$) of the FNN over time

$p_a = 0.001$, and a number k of automata such that $\rho = k/L^2 \simeq 0.2$. Hereby, ρ is called density. The algorithm for the simulation of the FNN (for m time steps) is shown in Algorithm 1.

Algorithm 1 Fluid Neural Network proposed in [5].

- 1: Initialize states: $S_i(t) := S_a$, $i = 1, \dots, k$
 - 2: **for** $t = 1, \dots, m$ **do**
 - 3: **for** all automata $i = 1, \dots, k$ **do**
 - 4: Calculate $a_i(t)$ (see Eq. 1)
 - 5: **if** $a_i(t) = 0$ **then**
 - 6: Draw a random number $p \in [0, 1]$
 - 7: **if** $p \leq p_a$ **then**
 - 8: $S_i(t) := S_a$
 - 9: $a_i(t) := 1$
 - 10: **end if**
 - 11: **end if**
 - 12: **end for**
 - 13: **for** all automata $i = 1, \dots, k$ **do**
 - 14: **if** $a_i(t) = 1$ **then**
 - 15: Decide randomly to stay at the current location or to move to one of the four neighbor sites
 - 16: **end if**
 - 17: **end for**
 - 18: Update state variables of all automata in parallel (see Eqs. 2 and 4)
 - 19: **end for**
-

III. ADAPTATION TO SENSOR NETWORKS: SYSTEM I

The main goal of this work is to study the feasibility of self-synchronized duty-cycling in mobile sensor networks inspired by the self-synchronization mechanism of real ant colonies as described in the previous section. We assume that sensor nodes are equipped with energy harvesting devices such as photovoltaic cells. Therefore, we not only aim at self-synchronization, but also at an *adaptive* duty-cycling mechanism depending on the available energy at any moment.

In the following we describe our system as an extension of the FNN described in the previous section. This system will be denoted by *System I*, in contrast to a modified system outlined in the following section. First, the environment of

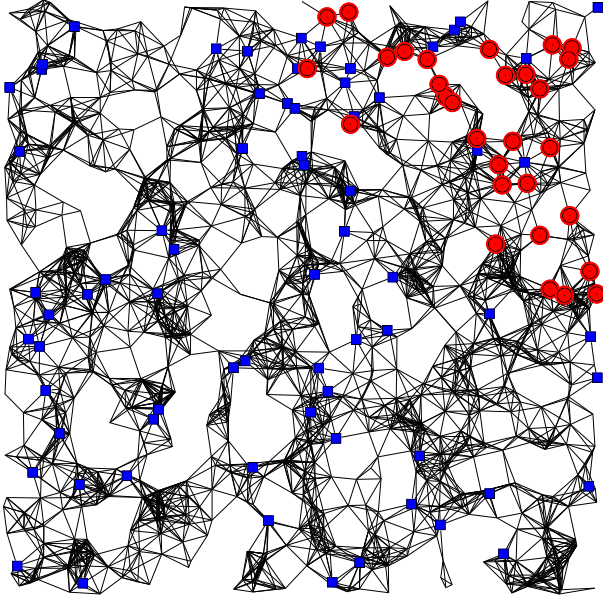


Fig. 2. r -disk graph with 1000 nodes and $r = 0.06$

the sensors will be modelled by an r -disk graph $G = (V, E)$ with n nodes, instead of an $L \times L$ grid. More specifically, we generated an r -disk graph by choosing for each of the n nodes randomly a location in $[0, 1] \times [0, 1]$. An edge (v_s, v_l) ($s < l$) is introduced between each two nodes v_s and v_l such that $d(v_s, v_l) \leq r$, where $d(v_s, v_l)$ denotes the Euclidian distance between the locations of v_s and v_l . At each time step t , each of the sensors resides on one of the nodes of G . The node on which a sensor i resides at time step t is henceforth denoted by $p_i(t)$. Moreover, active nodes may either stay where they are, or they may move to one of the neighbor nodes in graph G . Note that we chose this mobility model for simplicity reasons. Our system, however, could work with any other existing mobility model (see [16]). Finally, the density ρ is now calculated as k/n . Figure 2 shows an r -disk graph. Circles symbolize sensors that are active, and squares represent inactive sensors.

We used a simple energy model that can be explained as follows. Each sensor i has a battery. The battery level of sensor i at time t is denoted by $b_i(t) \in [0, 1]$. Hereby, $b_i(t) = 1$ corresponds to a full battery. Moreover, sensors are equipped with omni-directional antennas with a variable transmission radius. The transmission radius of sensor i at time step t is denoted by $r_i(t)$. With respect to the r -disk graph and the variable transmission radii, we redefined the set of neighbors $N_i(t)$ of a sensor i at time step t as follows:

$$N_i(t) := \{j \mid d(p_i(t), p_j(t)) \leq r_j(t)\} \quad (5)$$

In words, the neighbors of a sensor i are the sensors j that have a transmission radius such that i can receive their transmission. See Figure 3 for an example.

At each time step t , each sensor i consumes a certain amount of energy that depends on its current state. Sensors

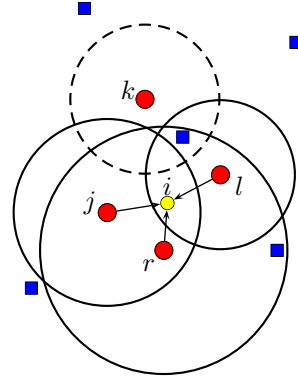


Fig. 3. Example for determining the neighborhood $N_i(t)$ of a sensor i (see Eq. 5). $N_i(t) = \{j, r, l\}$, because i is covered by the transmission radius of sensors j , r and l . Note that circles represent active sensors, whereas squares represent inactive ones.

that are *inactive* consume e_{sleep} of battery resources. Being inactive represents a state in which a sensor does nothing except for listening to incoming communication. In contrast, a sensor i that is *active* at time step t consumes an amount of energy that depends on its current transmission radius:

$$e_i(t) := e_{\text{awake}} \cdot (1 + r_i(t)) \quad (6)$$

Note that e_{sleep} and e_{awake} are parameters that depend strongly on the technical properties of the sensors used and on the tasks that sensors must execute.

Finally, the energy harvesting capabilities enable the sensors to produce a certain amount of energy at each time step t . As an example, we assume that sensors are equipped with solar cells, and that the light source at time step t has an intensity of $s(t) \in [0, 1]$. Hereby, $s(t) = 0$ corresponds to absolute darkness. Sensors can transform a fraction f of this intensity into energy:

$$e_i^{\text{harv}}(t) := f \cdot s(t) \quad (7)$$

The variable transmission radius of the sensors is a key component of our system. It will be used to adapt duty-cycling to changing energy availabilities. More specifically, the transmission radius of a sensor i at time t depends on the battery level $b_i(t)$ and in general also on the density ρ of the system. After experimenting with the system we decided for the following way of calculating the transmission radius:

$$r_i(t) := r_c + \xi(\rho) \cdot b_i(t)^2 \quad (8)$$

where r_c is a parameter and $\xi(\rho)$ is a factor that depends on ρ in the following way:

$$\xi(\rho) := \rho^2 - 0.35\rho + 0.125 \quad (9)$$

This function was chosen by simple interpolation such that the behaviour of the system is similar for any (reasonable) density ρ .

Finally, Algorithm 2 shows how the mobile sensor network is simulated.

Algorithm 2 Mobile Sensor Network Simulation

```
1: Initialize batteries:  $b_i(1) = 1, i = 1, \dots, k$ 
2: Initialize states:  $S_i(t) := S_a, i = 1, \dots, k$ 
3: for  $t = 1, \dots, m$  do
4:   for all sensors  $i = 1, \dots, k$  do
5:     Calculate  $a_i(t)$  (see Eq. 1)
6:     if  $a_i(t) = 0$  then
7:       Draw a random number  $p \in [0, 1]$ 
8:       if  $p \leq p_a$  then
9:          $S_i(t) := S_a$ 
10:         $a_i(t) := 1$ 
11:       end if
12:     end if
13:   end for
14:   for all sensors  $i = 1, \dots, k$  do
15:      $b_i(t) := b_i(t) + e_i^{\text{harv}}(t)$ 
16:      $r_i(t) := r_c + \xi(\rho) \cdot b_i(t)^2$ 
17:   end for
18:   Update all state variables in parallel (see Eqs. 2 and 4)
19:   for all sensors  $i = 1, \dots, k$  do
20:     if  $a_i(t) = 1$  then
21:       Decide randomly to stay at the current node or to
       move to one of the neighboring nodes in graph
        $G$ 
22:        $b_i(t) := b_i(t) - e_i(t)$ 
23:     else
24:        $b_i(t) := b_i(t) - e_{\text{sleep}}$ 
25:     end if
26:   end for
27: end for
```

A. Experimental Evaluation of System I

All experiments that we present in this paper have been obtained with 100 sensors moving on a randomly generated r -disk graph with 1000 nodes and $r = 0.06$. This corresponds to a density of $\rho = 0.1$. However, remember that due to Eq. 8, the sensor system will behave in a similar way also for other densities.

Concerning the length of a time step t , we have chosen it to be one minute. This means that a day consists of 1440 time steps. Moreover, we have used the following function for modelling the light intensities for energy harvesting:

$$s(t) = \begin{cases} 0, & \text{if } 0 \leq t < 420 \\ \frac{1 - \cos(\frac{t-420}{1140-420} \cdot 2\pi)}{2}, & \text{if } 420 \leq t < 1140 \\ 0, & \text{if } 1140 \leq t < 1440 \end{cases} \quad (10)$$

where $t = t \bmod 1440$. Note that this function can be seen as a model of the sun light intensities of a day (0:00 a.m.–24:00 p.m.). However, notice that our system should be able to adapt to any light intensity function. The one that we used is easily replaceable and should only be seen as an example.

Although this work focuses mainly on the swarm intelligence aspects and neglects many technicalities of real sensor networks, we still aim at a rather realistic setting of

the involved parameters. For example, we want to avoid a parameter setting such that the batteries are always nearly full. Neither do we want that energy harvesting is not enough to sustain the network even when energy is saved. In fact, for our experiments, we have decided that the capacities of the batteries should be able to keep all the energy harvested by the system in a single day, but no more. This goal can be achieved with a setting of $f = 0.0027$. For the derivation of this parameter setting see Appendix A. Finally, notice that each experiment described in the following consists of a simulation over five days: $t = 0, \dots, 7200$. In general, except for the first day, the system exhibits the same behaviour every day. The behaviour on the first day is slightly different, because all sensors start off with a full battery, being active. For each experiment we will show the representative behaviour of the system on the 3rd day.

A summary of the parameter settings used is shown in Table I. With these parameter settings we have conducted a first experiment. In Fig. 4(a) we show the behaviour of the system in terms of the measure $A(t)$ for each time step t (see Eq. 3). Note that the occurrence of peaks indicates self-synchronization. Additionally, Fig. 4(a) shows the evolution of the light intensity (denoted in the legend as *sun*) and the average battery level (denoted as *battery*). Clearly, during the night (that is, no light intensity) the average battery level decreases, and the peaks of system activity become smaller. This shows that the system adaptively tries to save energy. However, at all times we can notice self-synchronization among the sensors. During the day (that is, at the presence of light intensity) the batteries refill and the peaks of system activity grow. Moreover, when batteries are nearly full there is no need for switching to the inactive state and sensors are permanently active.

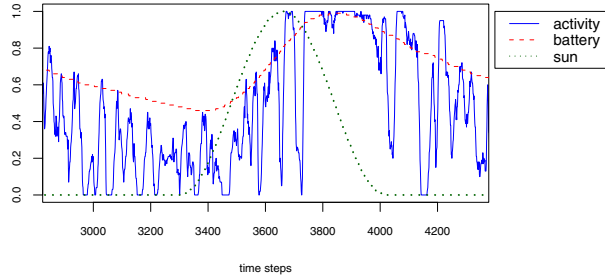
In a second experiment we wanted to study the possibility of having a system that behaves exactly the same as the system described in the first experiment, except of the fact that peaks are shorter and more frequent. In terms of duty-cycling this means that the frequency of changing between the active and the inactive state increases, while the time of being active (before switching again to the inactive state) decreases. In order to obtain this behaviour we increased the wake-up probability ($p_a = 0.01$) and reduced parameter g (see Eq. 2). Figure 4(b) shows that the behaviour is exactly as desired.

Finally, we wanted to study the behaviour of the system when no energy can be harvested. The simulation results are shown in Fig. 4(c). It is evident that with decreasing battery level, the system tries to save more and more energy, that is, the height of the activity peaks decreases constantly. After around two days of operation, the sensor network runs out of battery.

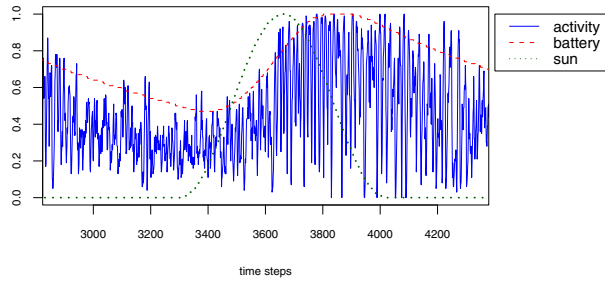
In addition, we have considered an energy consumption scheme in which sensors, in general, consume much more energy. This was done in order to analyze the generality and adaptability of the system. The new scheme is characterized

TABLE I
PARAMETER SETTINGS FOR SYSTEM I.

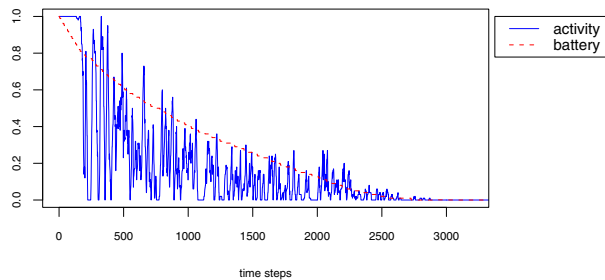
p_a	S_a	θ_{act}	r_c	f	e_{awake}	e_{sleep}	g
0.001	0.01	10^{-16}	0.05	0.0027	0.001	0.0002	0.1



(a) Behaviour of System I



(b) Behaviour of System I with $p_a = 0.01$ and $g = 0.01$



(c) Behaviour of System I, no energy harvesting

Fig. 4. Adaptive self-synchronization obtained with System I

by doubling the values of e_{sleep} and e_{awake} , that is, we chose $e_{sleep} = 0.0004$ and $e_{sleep} = 0.002$. Moreover, we increased substantially the increase in energy that must be spent when the transmission radius grows. More specifically, we exchanged Eq. 6 with the following one:

$$e_i(t) := e_{awake} \cdot (1 + 10 \cdot r_i(t)) \quad (11)$$

The behaviour of the resulting system is shown in Fig. 5. It can be seen that the system nicely adapts to the changed energy scheme by trying to save more energy. When comparing

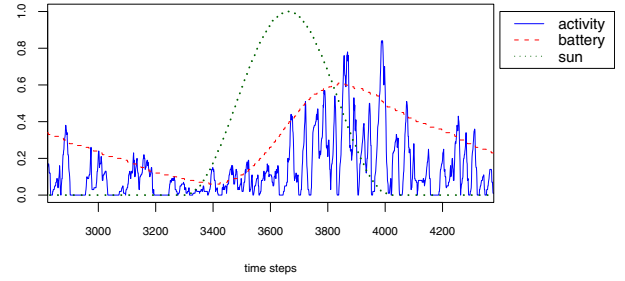


Fig. 5. Behaviour of System I, changed energy scheme

TABLE II
PARAMETER SETTINGS FOR SYSTEM II.

p_c^a	S_a	θ_{act}	r_c	f	e_{awake}	e_{sleep}	g
0.00005	0.01	10^{-16}	0.2	0.0027	0.001	0.0002	0.05

Fig. 5 with Fig. 4(a) it can be noticed that the activity peaks are generally lower, and the average battery level is lower as well.

IV. ADAPTATION TO SENSOR NETWORKS: SYSTEM II

The behaviour of System I as described in the previous section is such that with decreasing battery level, the height of the activity peaks decreases while the frequency of the peaks does not change. However, for some applications it might be useful to obtain a behaviour such that activity peaks always reach 1.0, that is, in each activity peak there is a time in which all sensors of the system are active, while energy saving is obtained by a changing peak frequency.

For this purpose we changed the system described in the previous section as follows. First, instead of a fixed activation probability p_a we introduced for each sensor i a variable activation probability that is computed as follows:

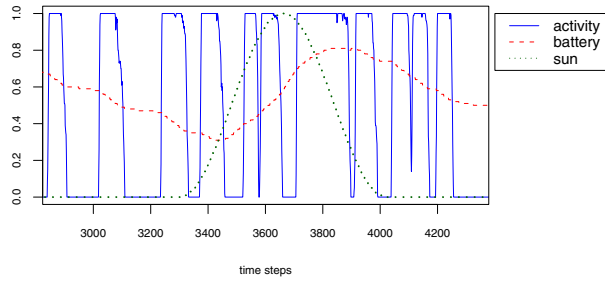
$$p_i^a(t) := p_c^a \cdot (1 + 100 \cdot b_i(t)^4), \quad (12)$$

where p_c^a is a base-probability that we set to 0.00005. Furthermore, we use a fixed transmission radius r_c instead of a variable one as in System I. A summary of the parameter settings for System II is presented in Table II.

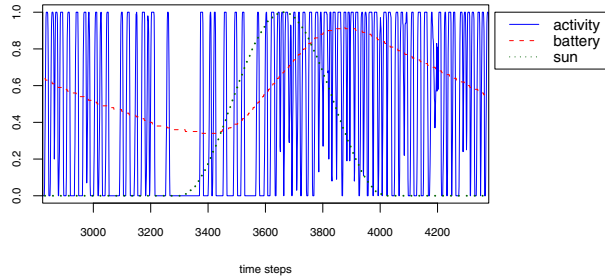
A. Experimental Evaluation of System II

We repeated the first three experiments conducted for System I also for System II. The simulation results for the parameter settings as shown in Table II are presented in Fig. 6(a). The graphic shows the desired behaviour of activity peaks of full height, whereas energy saving is achieved by changing the peak frequency.

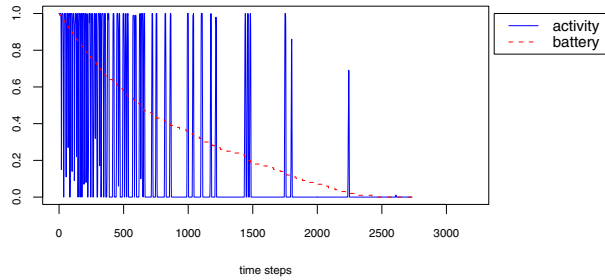
As in the previous section we wanted to study if the system can be changed such that the peaks are shorter and more frequent. For that purpose we changed the setting of parameters p_c^a and g as follows: $p_c^a = 0.0001$ and $g = 0.01$, that is, we increased the activation probability, and decreased



(a) Behaviour of System II



(b) Behaviour of System II with $p_c^a = 0.0001$ and $g = 0.01$



(c) Behaviour of System II, no energy harvesting

Fig. 6. Adaptive self-synchronization obtained with System II

the value of parameter g . The simulation results are shown in Fig. 6(b). The behaviour of the system basically remains the same, apart from the fact that peaks are now shorter and more frequent.

The third experiment is characterized by no light intensity at all, which prohibits energy harvesting. As it can be seen in Fig. 6(c), the frequency of the peaks decreases until the system runs out of energy after around one day and 14 hours.

Finally, we performed a first experiment for studying if our system also works in static sensor networks. This is an important issue, because the majority of sensor network applications deals with sensors without moving capabilities. For that purpose we equally distributed 100 sensors in the square $[0, 1] \times [0, 1]$. More specifically, we assigned the sensors to locations $(0.05 + i \cdot 0.1, 0.05 + j \cdot 0.1)$, $i, j = 0, \dots, 9$. System II with slightly changed parameter settings

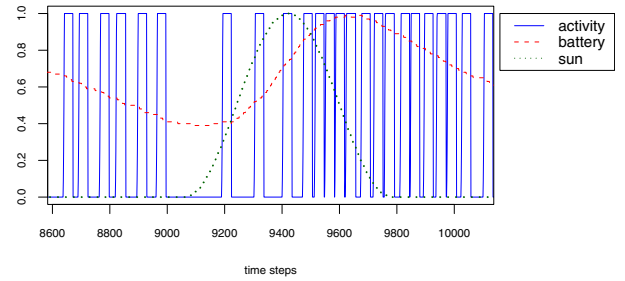


Fig. 7. Behaviour of System II for a static sensor network

($r_c = 0.15$ and $p_c^a = 0.00002$) obtained the behaviour that is shown in Fig. 7. Clearly, the system self-synchronizes and adapts to changing energy situations. However, the application to static networks must be studied in more detail before general claims can be made.

V. CONCLUSIONS

Inspired by the self-synchronization of activity periods in real ant colonies, we have proposed a mechanism for self-synchronized duty-cycling in mobile sensor networks with energy harvesting capabilities. Both duty-cycling and energy harvesting are important facilities for prolonging the lifetime of a wireless sensor network. We proposed two slightly different systems. The first one was characterized by a quite regular duty-cycling, whereas the height of the activity peaks depended on the current average battery level of the sensors. The second system exhibited always high activity peaks, whereas the frequency of these peaks changed depending on the current energy situation.

The study presented in this paper should be seen as a first feasibility study. We are aware of the fact that when designing a similar system for real sensor networks, there are strong technical constraints to be taken into account. In the future we plan, first, to better understand the system that we proposed. For example, it would be interesting to study if the system also works when simulated asynchronously. Furthermore, it would be very interesting to study the effects of different light intensities in different areas of the network at the same time. After dealing with these short term goals we certainly plan to adapt our system to real sensor networks.

ACKNOWLEDGEMENTS

This work was supported by the Integrated Action grant MEC HA2006-0127 (Germany/Spain), by the German Research Foundation (DFG) through project *Organisation and Control of Self-Organising Systems in Technical Compound* within SPP 1183, by the HFSP Research Grant *Optimization in natural systems: ants, bees and slime moulds* (2007), by grant TIN2007-66523 (FORMALISM) of the Spanish government, and by the EU project FRONTS (FP7-ICT-2007-1) funded by the European Commission under the FET Proactive Initiative *Pervasive Adaptation*. In addition, Christian Blum

acknowledges support from the *Ramón y Cajal* program of the Spanish Government of which he is a research fellow.

REFERENCES

- [1] D. Amit. *Modeling Brain Function*. Cambridge University Press, Cambridge, 1989.
- [2] K. Balachandran, J. H. Kang, and W. C. Lau. Adaptive sleeping and awakening protocol (ASAP) for energy efficient adhoc sensor networks. In *Proceedings of ICC 2005 – IEEE International Conference on Communications*, volume 2, pages 1068–1074. IEEE press, 2005.
- [3] B. J. Cole. Short-Term Activity Cycles in Ants: Generation of Periodicity by Worker Interaction. *American Naturalist*, 137(2):244, 1991.
- [4] B.J. Cole. The social behavior of *Leptothorax allardycei* (Hymenoptera, Formicidae): time budgets and the evolution of worker reproduction. *Behavioral Ecology and Sociobiology*, 18(3):165–173, 1986.
- [5] J. Delgado and R. V. Solé. Self-synchronization and Task Fulfilment in Ant Colonies. *Journal of Theoretical Biology*, 205(3):433–441, 2000.
- [6] S. Du, A. K. Saha, and D. B. Johnson. RMAC: A routing-enhanced duty-cycle MAC protocol for wireless sensor networks. In *Proceedings of INFOCOM 2007 – 26th IEEE International Conference on Computer Communications*, pages 1478–1486. IEEE press, 2007.
- [7] N. R. Franks and S. Bryant. Rhythmical patterns of activity within the nest of ants. *Chemistry and Biology of Social Insects*, pages 122–123, 1987.
- [8] N. R. Franks, S. Bryant, R. Griffiths, and L. Hemerik. Synchronization of the behaviour within nests of the antleptothorax acervorum (fabricius)-I. Discovering the phenomenon and its relation to the level of starvation. *Bulletin of Mathematical Biology*, 52(5):597–612, 1990.
- [9] J.M. Herbers. Social organisation in *Leptothorax* ants: within- and between-species patterns. *Psyche: A Journal of Entomology*, 90(4):361–386, 1983.
- [10] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, and V. Raghunathan. Adaptive duty cycling for energy harvesting systems. In *Proceedings of ISLPED 2006 – International Symposium on Low Power Electronics and Design*, pages 180–185. IEEE press, 2006.
- [11] X. Jiang, J. Polastre, and D. Culler. Perpetual environmentally powered sensor networks. In *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pages 463–468, 2005.
- [12] A. Kansal and MB Srivastava. An environmental energy harvesting framework for sensor networks. In *Low Power Electronics and Design, 2003. ISLPED'03. Proceedings of the 2003 International Symposium on*, pages 481–486, 2003.
- [13] Aman Kansal, Jason Hsu, Sadaf Zahedi, and Mani B. Srivastava. Power management in energy harvesting sensor networks. *Trans. on Embedded Computing Sys.*, 6(4):32, 2007.
- [14] O. Miramontes. Complexity and Behaviour in *Leptothorax* Ants. *Master Thesis*, 1992.
- [15] M. Rahimi, H. Shah, GS Sukhatme, J. Heideman, and D. Estrin. Studying the feasibility of energy harvesting in a mobile sensor network. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 1, 2003.
- [16] F. Schulz. *Algorithms for Sensor and Ad Hoc Networks*, volume 4621 of *Lecture Notes in Computer Science*, chapter Modeling Sensor and Ad Hoc Networks, pages 21–36. Springer Verlag, Berlin, Germany, 2007.
- [17] J.H. Sudd. *An Introduction to the Behaviour of Ants*. Edward Arnold. London, GB, 1967.
- [18] C. M. Vigorito, D. Ganesan, and A. G. Barto. Adaptive control of duty cycling in energy-harvesting wireless sensor networks. In *Proceedings of SECON 2007 – 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pages 21–30. IEEE press, 2007.
- [19] D. Wagner and R. Wattenhofer, editors. *Algorithms for Sensor and Ad Hoc Networks*, volume 4621 of *Lecture Notes in Computer Science*. Springer Verlag, Berlin, Germany, 2007.
- [20] F. Zhao and L. Guibas. *Wireless Sensor Networks—An Information Processing Approach*. The Morgan Kaufmann Series in Networking. Morgan Kaufmann Publishers, San Francisco, CA, 2004.

APPENDIX A

In the following we explain the setting of parameter f that controls the amount of energy that can be harvested. Remember that for our experiments we have decided that the capacities of the batteries should be able to keep all the energy harvested by the system in a single day, but no more. This goal can be formally expressed by the equation $f \cdot I = 1$, where f is the fraction of the light intensity that can be harvested and 1 represents the capacity of the batteries. Moreover,

$$I = \int_{\text{one_day}} s(t) . \quad (13)$$

Function $s()$ (as defined in Eq. 10) is different to 0 only in the interval $[420, 1140]$. Moreover, our system works with discrete time steps. Therefore:

$$I = \sum_{420}^{1440} \frac{1 - \cos\left(\frac{t-420}{1140-420} \cdot 2\pi\right)}{2} \quad (14)$$

Solving the calculus we get that $I = 360$. With Eq. 13, we obtain

$$f = \frac{1}{I} = \frac{1}{360} \approx 0.002778 . \quad (15)$$

Note that when replacing function $s()$ by another one, the value of f must be adjusted with respect to the new function. Also notice that we can modify the value of f , for example, for the case that the capacity of a battery is twice the energy that can be obtained from the sun in one day, or similar.